

A generalized fault-detection software reliability model subject to random operating environments

Hoang Pham¹ 

Received: 9 March 2016 / Accepted: 10 March 2016 / Published online: 22 April 2016
© The Author(s) 2016. This article is published with open access at Springerlink.com

Abstract Many software reliability growth models (SRGMs) have been developed in the past three decades to estimate software reliability measures such as the number of remaining faults and software reliability. The underlying common assumption of many existing models is that the operating environment and the developing environment are the same. This is often not the case in practice because the operating environments are usually unknown due to the uncertainty of environments in the field. In this paper, we develop a generalized software reliability model incorporating the uncertainty of fault-detection rate per unit of time in the operating environments. A logistic fault-detection software reliability model is derived. Examples are included to illustrate the goodness of fit of the proposed model and existing nonhomogeneous Poisson process (NHPP) models based on a set of failure data. Three goodness-of-fit criteria, such as mean square error, predictive power, and predictive ratio risk are used as an example to illustrate model comparisons. The results show that the proposed logistic fault-detection model fit significantly better than other existing NHPP models based on all three goodness-of-fit criteria.

Keywords Non-homogeneous Poisson process · Software reliability growth model · Mean square error · Predictive ratio risk · Predictive power · Logistic fault-detection rate

1 Introduction

Many existing NHPP software reliability models [1–28] have been used through the fault intensity rate function and the mean value functions $m(t)$ within a controlled testing environment to estimate reliability metrics such as the number of residual faults, failure rate, and reliability of the software. Generally, these models are applied to the software testing data and then used to make predictions on the software failures and reliability in the field. In other words, the underlying common assumption of such models is that the operating environment and the developing environment are about the same [21, 27]. The operating environments in the field for the software, in reality, are quite different. The randomness of the operating environments will affect software failure and software reliability in an unpredictable way.

Estimating software reliability in the field is important, yet a difficult task. Usually, software reliability models are applied to system test data with the hope of estimating the failure rate of the software in user environments. Teng and Pham [3] discussed a generalized model that captures the uncertainty of the environment and its effects upon the software failure rate. Other researchers [8, 19–21, 24, 28] have also developed reliability and cost models incorporating both testing phase and operating phase in the software development cycle for estimating the reliability of software systems in the field. Pham et al. [26] recently discussed a new logistic software reliability model where the fault-detection rate per unit time follows a three-parameter logistic function. They did not take into consideration of the uncertainty of operating environment. Pham [27] recently also developed a software reliability model with Vtub-shaped fault-detection rate subject to the uncertainty of operating environments.

In this paper, we discuss a new generalized software reliability model subject to the uncertainty of operating envi-

✉ Hoang Pham
hopham@rci.rutgers.edu

¹ Department of Industrial and Systems Engineering, Rutgers University, Piscataway, NJ 08854, USA

ronments. The explicit solution of the generalized model and a specific model with a logistic fault-detection rate function IS derived in Sect. 2. Model analysis and results are discussed in Sect. 3 to illustrate the performance of THE proposed model and compare it to several common existing NHPP models based on three existing criteria such as mean square error, predictive power and predictive ratio risk from a set of software failure data. Section 4 concludes the paper.

Notation

- $m(t)$ Expected number of software failures detected by time t .
 N Expected number of faults that exist in the software before testing.
 $b(t)$ Time-dependent fault-detection rate per unit of time.

2 A generalized NHPP model with random operating environments

Many existing NHPP models assume that failure intensity is proportional to the residual fault content. A generalized mean value function $m(t)$ with the uncertainty of operating environments [27] can be obtained by solving the following defined differential equation:

$$\frac{dm(t)}{dt} = \eta b(t)[N - m(t)], \quad (1)$$

where η is a random variable that represents the uncertainty of the system detection rate in the operating environments with a probability density function g . The solution for the mean value function $m(t)$, where the initial condition $m(0) = 0$, is given by [27]:

$$m(t) = \int_{\eta} N \left(1 - e^{-\eta \int_0^t b(x) dx} \right) dg(\eta). \quad (2)$$

Based on the above equation, in this study we assume that the random variable η has a generalized probability density function g with two parameters $\alpha \geq 0$ and $\beta \geq 0$, so that the mean value function from Eq. (2) can be obtained in the general form below:

$$m(t) = N \left(1 - \frac{\beta}{\beta + \int_0^t b(s) ds} \right)^{\alpha}, \quad (3)$$

where $b(t)$ is the fault-detection rate per fault per unit of time.

Depending on how elaborate a model one wishes to obtain, one can use $b(t)$ to yield more complex or less complex

analytic solutions for the function $m(t)$. Various $b(t)$ reflects various assumptions of the growth processes. In this paper, we use the time-dependent three-parameter logistic function (or “S-shaped” curve) below to describe the fault-detection rate per fault per unit of time in the software system:

$$b(t) = \frac{c}{1 + ae^{-bt}} \quad \text{for } a \geq 0, b \geq 0, c > 0. \quad (4)$$

The characteristic “S-shaped” curve of a logistic function shows that the initial exponential growth is followed by a period in which growth slows, then increases rapidly, and eventually levels off, approaching (but never attaining) a maximum upper limit. Substituting the three-parameter logistic function $b(t)$ from Eq. (4) into Eq. (3), we can obtain the expected number of software failures detected by time t subject to the uncertainty of the environments as follows:

$$m(t) = N \left(1 - \frac{\beta}{\beta + \left(\frac{c}{b} \right) \ln \left(\frac{a+e^{bt}}{1+a} \right)} \right)^{\alpha}. \quad (5)$$

Table 1 summarizes the proposed model and several existing well-known NHPP models with different mean value functions.

3 Model analysis and results

3.1 Some existing criteria

There are several existing goodness-of-fit criteria. In this study, we apply three common criteria for model performance and comparisons. They are: the mean square error, the predictive ratio risk, and the predictive power. Below is a brief description of the criteria.

The mean square error (MSE) measures the deviation between the predicted values with the actual observation and is defined as:

$$\text{MSE} = \frac{\sum_{i=1}^n (\hat{m}(t_i) - y_i)^2}{n - k}, \quad (6)$$

where n and k are the number of observations and number of parameters in the model, respectively.

The predictive ratio risk (PRR) measures the distance of model estimates from the actual data against the model estimate and is defined as [17]:

$$\text{PRR} = \sum_{i=1}^n \left(\frac{\hat{m}(t_i) - y_i}{\hat{m}(t_i)} \right)^2, \quad (7)$$

Table 1 A summary of new and existing software reliability models

Model	$m(t)$
Goel–Okumoto (G–O)	$m(t) = a(1 - e^{-bt})$
Delayed S-shaped	$m(t) = a(1 - (1 + bt)e^{-bt})$
Inflection S-shaped	$m(t) = \frac{a(1 - e^{-bt})}{1 + \beta e^{-bt}}$
Yamada imperfect debugging I	$m(t) = a[1 - e^{-bt}][1 - \frac{\alpha}{b}] + \alpha at$
Pham inflexion [26]	$m(t) = N \left(1 - \frac{1}{\left(\frac{\beta + e^{bt}}{1 + \beta} \right)^{\frac{a}{b}}} \right)$
PNZ model	$m(t) = \frac{a}{1 + \beta e^{-bt}} ([1 - e^{-bt}][1 - \frac{\alpha}{b}] + \alpha t)$
Pham–Zhang model	$m(t) = \frac{1}{1 + \beta e^{-bt}} \left((c + a)(1 - e^{-bt}) - \frac{ab}{b - \alpha} (e^{-\alpha t} - e^{-bt}) \right)$
Dependent-parameter model (Model 1) [18]	$m(t) = \alpha(1 + \gamma t)(\gamma t + e^{-\gamma t} - 1)$
Dependent-parameter model with $m(t_0) \neq 0$ (Model 1a) [18]	$m(t) = m_0 \left(\frac{\gamma t + 1}{\gamma t_0 + 1} \right) e^{-\gamma(t - t_0)} + \alpha(\gamma t + 1) [\gamma t - 1 + (1 - \gamma t_0)e^{-\gamma(t - t_0)}]$
Vtub-shaped fault-detection rate model [27]	$m(t) = N \left(1 - \left(\frac{\beta}{\beta + a^{\frac{1}{b}} - 1} \right)^{\alpha} \right)$
Logistic fault-detection model (new model)	$m(t) = N \left(1 - \frac{\beta}{\beta + \left(\frac{e}{b} \right) \ln \left(\frac{a + e^{bt}}{1 + a} \right)} \right)^{\alpha}$

where y_i is the total number of failures observed at time t_i according to the actual data and $\hat{m}(t_i)$ is the estimated cumulative number of failures at time t_i for $i = 1, 2, \dots, n$.

The predictive power (PP) measures the distance of model estimates from the actual data against the actual data, is as follows:

$$PP = \sum_{i=1}^n \left(\frac{\hat{m}(t_i) - y_i}{y_i} \right)^2. \quad (8)$$

For all these three criteria—MSE, PRR, and PP—the smaller the value, the better the model fits, relative to other models run on the same data set.

3.2 Software failure data

A set of system test data were provided in [2, page 149] which is referred to as Phase 2 data set and is given in Table 2. In this data set, the number of faults detected in each week of testing is found and the cumulative number of faults since the start of testing is recorded for each week. This data set provides the cumulative number of faults by each week up to 21 weeks. We use a Matlab software to perform the calculations for LSE parameter estimates.

3.3 Model results and comparison

Table 3 summarizes the result of the estimated parameters of all 11 models as shown in Table 1 using the least square

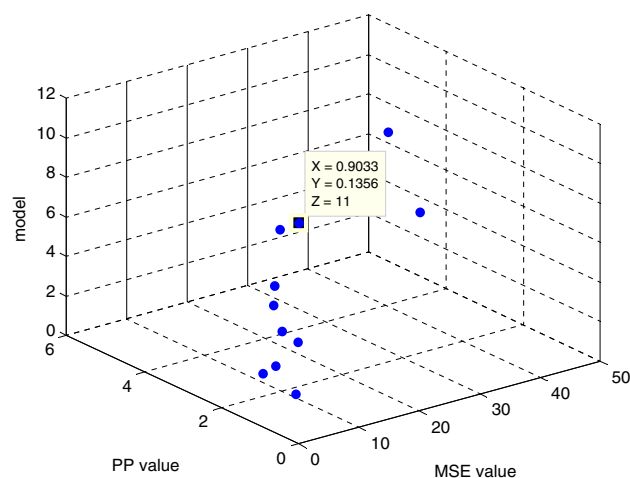
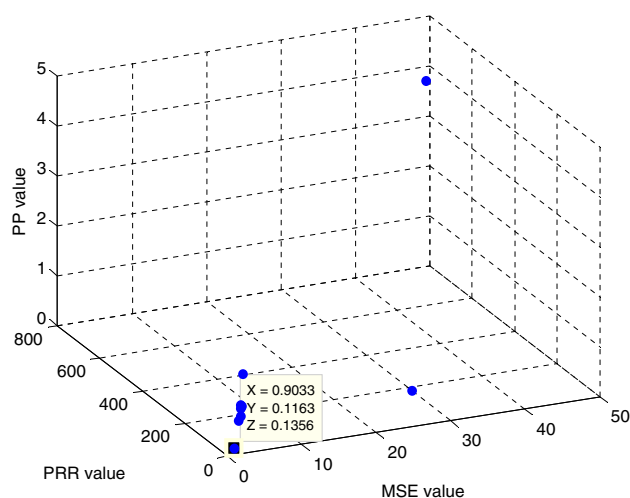
Table 2 Phase 2 system test data [2]

Week index	Exposure time (cum. system test hours)	Fault	Cum. fault
1	416	3	3
2	832	1	4
3	1248	0	4
4	1664	3	7
5	2080	2	9
6	2496	0	9
7	2912	1	10
8	3328	3	13
9	3744	4	17
10	4160	2	19
11	4576	4	23
12	4992	2	25
13	5408	5	30
14	5824	2	32
15	6240	4	36
16	6656	1	37
17	7072	2	39
18	7488	0	39
19	7904	0	39
20	8320	3	42
21	8736	1	43

Table 3 Model parameter estimation and comparison criteria

Model name	LSEs	MSE	PRR	PP
1. G–O model	$\hat{a} = 98,295$ $\hat{b} = 5.2 \times 10^{-8}$	6.61	0.69	1.10
2. Delayed S-shaped	$\hat{a} = 62.3$ $\hat{b} = 2.85 \times 10^{-4}$	3.27	44.27	1.43
3. Inflection S-shaped	$\hat{a} = 46.6$ $\hat{b} = 5.78 \times 10^{-4}$ $\hat{\beta} = 12.20$	1.87	5.94	0.90
4. Yamada imperfect debugging model	$\hat{a} = 1.5$ $\hat{b} = 1.1 \times 10^{-3}$ $\hat{\alpha} = 3.8 \times 10^{-3}$	4.98	4.30	0.81
5. Pham inflexion	$N = 45.8270$ $a = 0.2961$ $b = 0.2170$ $\beta = 13.6298$	1.5108	3.1388	0.6800
6. PNZ model	$\hat{a} = 45.99$ $\hat{b} = 6.0 \times 10^{-4}$ $\hat{\alpha} = 0$ $\hat{\beta} = 13.24$	1.99	6.83	0.96
7. Pham–Zhang model	$\hat{a} = 0.06$ $\hat{b} = 6.0 \times 10^{-4}$ $\hat{\alpha} = 1.0 \times 10^{-4}$ $\hat{\beta} = 13.2$ $\hat{c} = 45.9$	2.12	6.79	0.95
8. Dependent parameter model (Model 1)	$\hat{\alpha} = 3.0 \times 10^{-6}$ $\hat{\gamma} = 0.49$	43.69	601.34	4.53
9. Dependent parameter model with $m(t_0) \neq 0, t_0 \neq 0$ (Model 2)	$\hat{\alpha} = 890,996$ $\hat{\gamma} = 1.2 \times 10^{-6}$ $t_0 = 832$ $m_0 = 4$	24.79	1.14	0.73
10. Vtub-shaped fault-detection rate model (Pham Vtub model)	$\hat{N} = 43.25$ $\hat{a} = 2.662$ $\hat{b} = 0.196$ $\hat{\alpha} = 4.040$ $\hat{\beta} = 35.090$	1.80	2.06	0.77
11. New model	$\hat{N} = 50.56$ $\hat{a} = 10,000$ $\hat{b} = 0.7364$ $\hat{c} = 0.2038$ $\hat{\alpha} = 0.2554$ $\hat{\beta} = 1.778$	0.9033	0.1163	0.1356

estimation (LSE) technique and their criteria (MSE, PRR and PP) values. The coordinates X , Y and Z represent the MSE, PP, and the model, respectively, as shown in Fig. 1. Figure 2 shows the MSE, PRR and PP values of all the models. As we can see from Table 3, the new model has the smallest MSE, PP and PRR values. The plots in Fig. 3 illustrate the expected

**Fig. 1** A three-dimensional (X, Y, Z) represents (MSE, PP, model), respectively**Fig. 2** A three-dimensional (X, Y, Z) represents (MSE, PRR, PP), respectively

number of failures detected versus testing time t . Table 4 includes the rank of each model based on each criteria.

The new model (model 11) as shown in Fig. 2 and Table 4 provides the best fit based on the MSE, PRR and PP criteria. Obviously, further work in broader validation of this conclusion is needed using other data sets as well as considering other comparison criteria.

4 Conclusion

In this paper, we present a new general software reliability model incorporating the uncertainty of the operating environments. The explicit mean value function solution of the proposed model for a logistic fault-detection rate is presented. The results of the estimated parameters of the pro-

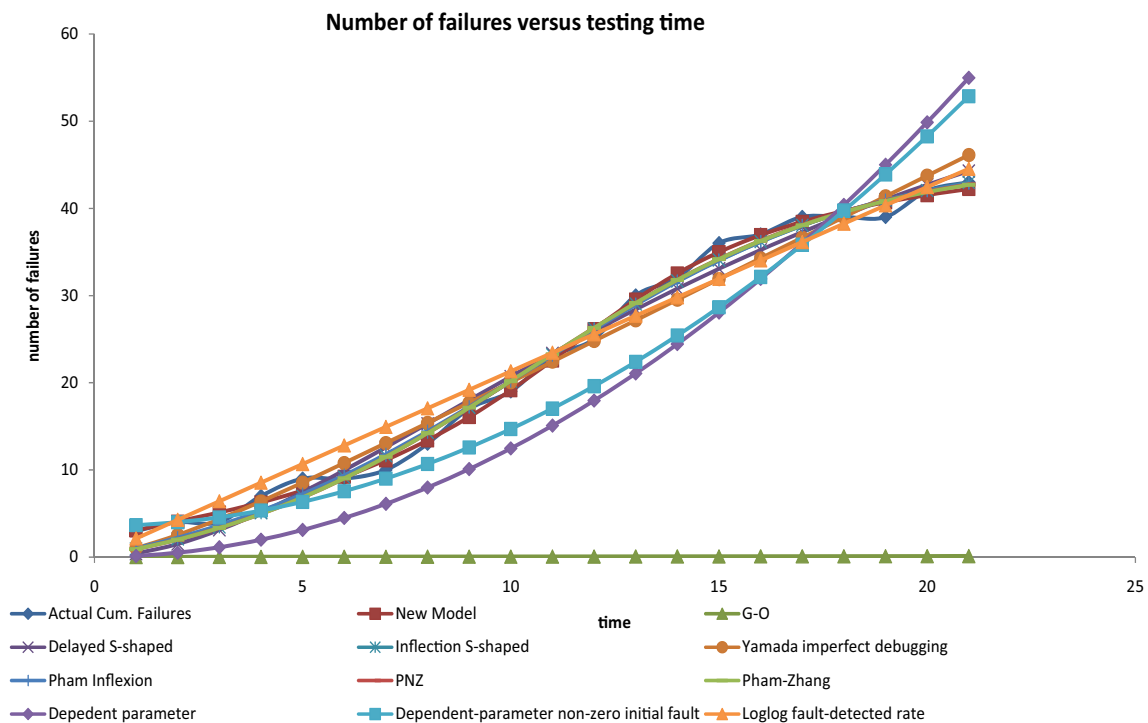


Fig. 3 Number of failures vs. testing time for all models based on Phase 2 data

Table 4 Parameter estimation and model comparison

Model/criteria	MSE (rank)	PRR (rank)	PP (rank)
1. G–O model	6.61 (9)	0.69 (2)	1.10 (9)
2. Delayed S-shaped	3.27 (7)	44.27 (10)	1.43 (10)
3. Inflection S-shaped	1.87 (4)	5.94 (7)	0.90 (6)
4. Yamada imperfect debugging model	4.98 (8)	4.30 (6)	0.81 (5)
5. Pham inflexion	1.5108 (2)	3.1388 (5)	0.6800 (2)
6. PNZ model	1.99 (5)	6.83 (9)	0.96 (8)
7. Pham–Zhang model	2.12 (6)	6.79 (8)	0.95 (7)
8. Dependent-parameter model	43.69 (11)	601.34 (11)	4.53 (11)
9. Dependent-parameter model with $m(t_0) \neq 0$, $t_0 \neq 0$	24.79 (10)	1.14 (3)	0.73 (3)
10. Pham Vtub model	1.80 (3)	2.06 (4)	0.77 (4)
11. Logistic fault-detection model (new model)	0.9033 (1)	0.1163 (1)	0.1356 (1)

posed model and several NHPP models are discussed. The results show that the proposed logistic fault-detection model fits significantly better than all the existing NHPP models studied in this paper based on all MSE, PRR and PP criteria. Obviously, further work in broader validation of this conclusion is needed using other data sets as well as considering other comparison criteria.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Goel, A.L., Okumoto, K.: Time-dependent fault-detection rate model for software and other performance measures. *IEEE Trans. Reliab.* **28**, 206–211 (1979)
- Pham, H.: *System Software Reliability*. Springer, Berlin (2006)
- Teng, X., Pham, H.: A new methodology for predicting software reliability in the random field environments. *IEEE Trans. Reliab.* **55**(3), 458–468 (2006)
- Ohba, M.: Inflexion S-shaped software reliability growth models. In: Osaki, S., Hatoyama, Y. (eds.) *Stochastic Models in Reliability Theory*, pp. 144–162. Springer, Berlin (1984)
- Pham, H.: Software reliability assessment: imperfect debugging and multiple failure types in software development. EG&G-RAAM-10737, Idaho National Engineering Laboratory (1993)
- Pham, H.: A software cost model with imperfect debugging, random life cycle and penalty cost. *Int. J. Syst. Sci.* **27**(5), 455–463 (1996)
- Ohba, M., Yamada, S.: S-shaped software reliability growth models. In: *Proceedings of the 4th International Conference on Reliability and Maintainability*, pp. 430–436 (1984)
- Teng, X., Pham, H.: A software cost model for quantifying the gain with considerations of random field environments. *IEEE Trans. Comput.* **53**(3), 380–384 (2004)
- Zhang, X., Teng, X., Pham, H.: Considering fault removal efficiency in software reliability assessment. *IEEE Trans. Syst. Man Cybern. Part A* **33**(1), 114–120 (2003)
- Pham, H., Zhang, X.: NHPP software reliability and cost models with testing coverage. *Eur. J. Oper. Res.* **145**, 443–454 (2003)
- Pham, H., Nordmann, L., Zhang, X.: A general imperfect software debugging model with s-shaped fault detection rate. *IEEE Trans. Reliab.* **48**(2), 169–175 (1999)
- Pham, H., Zhang, X.: An NHPP software reliability model and its comparison. *Int. J. Reliab. Qual. Saf. Eng.* **4**(3), 269–282 (1997)
- Pham, L., Pham, H.: Software reliability models with time-dependent hazard function based on Bayesian approach. *IEEE Trans. Syst. Man Cybern. Part A* **30**(1), 25–35 (2000)
- Yamada, S., Ohba, M., Osaki, S.: S-shaped reliability growth modeling for software fault detection. *IEEE Trans. Reliab.* **12**, 475–484 (1983)
- Yamada, S., Osaki, S.: Software reliability growth modeling: models and applications. *IEEE Trans. Softw. Eng.* **11**, 1431–1437 (1985)
- Yamada, S., Tokuno, K., Osaki, S.: Imperfect debugging models with fault introduction rate for software reliability assessment. *Int. J. Syst. Sci.* **23**(12), 2253–2264 (1992)
- Pham, H., Deng, C.: Predictive-ratio risk criterion for selecting software reliability models. In: *Proceedings of the Ninth International Conference on Reliability and Quality in Design*, August 2003
- Pham, H.: An imperfect-debugging fault-detection dependent-parameter software. *Int. J. Autom. Comput.* **4**(4), 325–328 (2007)
- Zhang, X., Pham, H.: Software field failure rate prediction before software deployment. *J. Syst. Softw.* **79**, 291–300 (2006)
- Sgarbossa, F., Pham, H.: A cost analysis of systems subject to random field environments and reliability. *IEEE Trans. Syst. Man Cybern. Part C* **40**(4), 429–437 (2010)
- Pham, H.: A software reliability model with vtub-shaped fault-detection rate subject to operating environments. In: *Proceedings of the 19th ISSAT International Conference on Reliability and Quality in Design*, Hawaii, August 2013
- Kapur, P.K., Pham, H., Aggarwal, A.G., Kaur, G.: Two dimensional multi-release software reliability modeling and optimal release planning. *IEEE Trans. Reliab.* **61**(3), 758–768 (2012)
- Kapur, P.K., Pham, H., Anand, S., Yadav, K.: A unified approach for developing software reliability growth models in the presence of imperfect debugging and error generation. *IEEE Trans. Reliab.* **60**(1), 331–340 (2011)
- Persona, A., Pham, H., Sgarbossa, F.: Age replacement policy in random environment using systemability. *Int. J. Syst. Sci.* **41**(11), 1383–1397 (2010)
- Xiao, X., Dohi, T.: Wavelet shrinkage estimation for non-homogeneous Poisson process based software reliability models. *IEEE Trans. Reliab.* **60**(1), 211–225 (2011)
- Pham, H., Pham, D.H., Pham Jr., H.: A new mathematical logistic model and its applications. *Int. J. Inf. Manag. Sci.* **25**, 79–99 (2014)
- Pham, H.: A new software reliability model with Vtub-shaped fault-detection rate and the uncertainty of operating environments. *Optimization* **63**(10), 1481–1490 (2014)
- Pham, H.: Loglog fault-detection rate and testing coverage software reliability models subject to random environments. *Vietnam J. Comput. Sci.* **1**(1), 39–45 (2014)